

Dockerbank II Praktische Übung: Komplexbeispiel *tranSMART*

Linux root:docker | SSH (putty): localhost:22
 Postgres postgres:docker | PgAdmin3: localhost:5432 | `psql -h transmart-db -U <username> -d postgres`
 Dateien: Übungen: /docker/exercise | Lösungen: /docker/solution | Cheatsheet: /docker/dockercheatsheet.txt
 tranSMART admin:admin | http://localhost:8080/transmart

Nr.	Aufgabe
1	Erstellung einer tranSMART-Datenbank
1.1	Verbinden Sie sich mit der laufenden Workshop-VM mit Hilfe von putty (http://www.putty.org) über die Adresse localhost und Port 22. Melden Sie sich mit Benutzernamen/Passwort root/docker an.
1.2	Navigieren Sie in den Ordner <code>cd /docker/exercise/part1</code> .
1.3	Lesen sie die Beschreibungen der zwei im eigenen Docker-Compose zu nutzenden Container https://hub.docker.com/r/tmfev/transmart-db/ sowie https://hub.docker.com/r/tmfev/transmart-db-data/ . Editieren Sie das DockerCompose-File (<code>vim docker-compose.yml</code>), um die bereits eingetragenen Services <code>transmartdbdata</code> und <code>transmartdb</code> zu konfigurieren und zu verbinden.
1.4	Kompilieren und führen Sie Docker-Compose aus: <code>docker-compose up</code>
1.5	Mit Hilfe von PgAdmin3 (https://www.pgadmin.org) oder <code>psql</code> (direkt auf der VM: <code>psql -h transmart-db -U postgres -d postgres</code>) können Sie sich auf der laufenden Instanz von <code>transmart-db</code> auf der Adresse localhost und über den von Ihnen freigegeben Port einloggen. Läuft die Datenbank?
1.6	Stoppen Sie den laufenden Container (<code>strg+c</code>) und löschen Sie ihn anschließend: <code>docker-compose rm -f</code>
2	Installation eines tranSMART-Upload-Tools und upload von Beispielstudien
2.1	Navigieren Sie in den Ordner <code>cd /docker/exercise/part2</code> .
2.2	Lesen Sie Beschreibung des Containers https://hub.docker.com/r/tmfev/transmart-db-manage/ . Fügen Sie die fehlenden Einträge für <code>image</code> , <code>depends_on</code> , <code>links</code> und benötigte Variables des <code>environment</code> in YAML-Notation hinzu.
2.3	Kompilieren und führen Sie Ihren Docker-Compose aus: <code>docker-compose up</code>
2.4	Verbinden sie sich mit PgAdmin3 zu Ihrer Datenbank und navigieren Sie zu Datenbanken/transmart/Schemata/i2b2metadata/Tabellen/i2b2 und lassen Sie sich die Daten anzeigen (Rechts-Klick auf „i2b2“->“Daten anzeigen“->“Die obersten 100 Zeilenzeigen“). Psql: <code>psql -h transmart-db -U postgres -d transmart -c „select * from i2b2metadata.i2b2 limit 100;“</code>
2.5	Stoppen Sie den laufenden Container (<code>strg+c</code>) und löschen Sie ihn anschließend: <code>docker-compose rm -f</code>
3	Installation der tranSMART-Application
3.1	Navigieren Sie in den Ordner <code>cd /docker/exercise/part3</code> .
3.2	Lesen Sie Beschreibung des Containers https://hub.docker.com/r/tmfev/transmart-app/ . Fügen Sie die fehlenden Einträge für <code>image</code> , <code>depends_on</code> , <code>links</code> , <code>ports</code> und benötigte Variables des <code>environment</code> in YAML-Notation hinzu und ergänzen die neue Option <code>cap_add</code> .
3.3	Kompilieren und führen Sie Ihren Docker-Compose aus: <code>docker-compose up</code>
3.4	Öffnen sie einen Browser und rufen sie die Adresse http://localhost:8080/transmart auf. Loggen sie sich mit admin/admin in Ihre tranSMART-Instanz ein. In der oberen rechten Ecke können sie über den Link <code>Analyze</code> die geladenen Beispielstudien nutzen.
2.5	Stoppen Sie den laufenden Container (<code>strg+c</code>) und löschen Sie ihn anschließend: <code>docker-compose rm -f</code>
4	Lösung
4.1	Unter <code>cd /docker/solution</code> befindet sich im Ordern <code>transmartCompose</code> die korrekte Lösung dieser Übung.
4.2	Vergleichen Sie die Lösung mit Ihrer.
4.3	Kompilieren und führen Sie den TMF tranSMART Compose-Container aus: <code>docker-compose up</code>